

LISTING OF THE CLAIMS

1. (Currently Amended) A method for customizing an application comprising the steps of:
identifying a selection of components to be deployed to form the customized application;
specifying within said selection of components, points of variability for the variable establishment of configuration parameters for the customized application which can be assigned values when deploying said selection of components;
persisting said identified selection of components and said specified points of variability in a template; and,
at a subsequent time, processing said template to deploy said identified selection of components, to prompt for values to be assigned to said points of variability, to configure said identified selection of components with said values at said points of variability thereby producing the customized application.
2. (Original) The method of claim 1, further comprising the steps of:
determining pre-requisite resources required by the customized application; and,
further persisting said determined pre-requisite resources in said template.
3. (Original) The method of claim 1, wherein said identifying step comprises the step of identifying a selection of portlets disposed within a portal environment to form the customized application.

4. (Currently Amended) A system for customizing an application comprising:
 - a template interface prototyped both for exposing content included within components implementing said template creation interface, and also for deploying said components implementing said template;
 - a template creation process coupled to said template interface, said template creation process having a configuration for writing a template persisting both references to a selection of components to be deployed to form the customized application and also points of variability for the variable establishment of configuration parameters for the customized application which can be assigned values when deploying said selection of components; and,
 - a template deployment process programmed to produce the customized application based upon a template written by said template creation process.
5. (Original) The system of claim 4, wherein said template creation process comprises a further configuration for persisting at least one reference to a pre-requisite resource required by the customized application to operate properly.
6. (Original) The system of claim 4, wherein said template deployment process further comprises a view for prompting an end user for values for said points of variability persisted in said template.
7. (Original) The system of claim 4, wherein said selection of components comprise portlet components disposed within a portal environment.

8. (Currently Amended) A method for producing a template describing a customized configuration of an application, the method comprising the steps of:

selecting a set of components to be deployed in the customized configuration;
specifying within said set, points of variability for the variable establishment of configuration parameters for the customized application which can vary when in each deployment of the customized configuration;
identifying dependencies among said set of components, said identified dependencies forming a hierarchy of components to be deployed in the customized configuration; and,
writing a template both enumerating said specified points of variability, and also listing said selected set of components while preserving said hierarchy within said listing in said template.

9. (Original) The method of claim 8, wherein said identifying step comprises the steps of:

descending said set of components in a top-down manner; and,
arranging said set of components in a hierarchy of dependencies during said descent.

10. (Original) The method of claim 8, wherein said writing step further comprises the step of writing within said template at least one reference to a pre-requisite resource.

11. (Original) The method of claim 8, wherein said listing step comprises the steps of:

accessing a template interface implemented by said set of components to permit access to content encapsulated by said set of components; and,

where a component within said set of components has not implemented said template interface, accessing said encapsulated content through a proxy to said component within said set of components.

12. (Original) The method of claim 8, wherein said preserving step comprises the step of nesting component references in said listing in said template according to relative positions in said hierarchy of said set of components.

13. (Currently Amended) A method for customizing an application, the method comprising the steps of:

loading a template describing a customized configuration of the application;
locating within said template a set of components arranged in a hierarchy of dependent components which are to be deployed in a customized configuration;
further locating within said template, a listing of points of variability for the variable establishment of configuration parameters for the application which can vary when in each deployment of the customized configuration;
deploying said set of components;
prompting an end user for values for said points of variability and applying said values to said points of variability; and,
returning control of the customized configuration to said end user.

14. (Original) The method of claim 13, wherein said locating step comprises the steps of:
descending said hierarchy recursively; and,

deploying each of component in said hierarchy beginning with components in said set which are not dependent upon any other component in said hierarchy, and continuing through to at least one root component which is dependent upon all other components disposed below said at least one root component in said hierarchy.

15. (Original) The method of claim 13, further comprising the steps of:

identifying within said template at least one reference to a pre-requisite resource; and,
deploying each pre-requisite resource identified within said template.

16. (Original) The method of claim 13, wherein said deploying step comprises the steps of:

accessing a template interface implemented by said set of components to permit access to content encapsulated by said set of components; and,

where a component within said set of components has not implemented said template interface, accessing said encapsulated content through a proxy to said component within said set of components.

17. Cancelled.

18. Cancelled.

19. Cancelled.

20. Cancelled.

21. (Currently Amended) A machine readable storage having stored thereon a computer program for producing a template describing a customized configuration of an application, the computer program comprising a routine set of instructions for causing the machine to perform the steps of:

selecting a set of components to be deployed in the customized configuration;
specifying within said set, points of variability for the variable establishment of configuration parameters for the customized application which can vary when in each deployment of the customized configuration;

identifying dependencies among said set of components, said identified dependencies forming a hierarchy of components to be deployed in the customized configuration; and,

writing a template both enumerating said specified points of variability, and also listing said selected set of components while preserving said hierarchy within said listing in said template.

22. (Original) The machine readable storage of claim 21, wherein said identifying step comprises the steps of:

descending said set of components in a top-down manner; and,
arranging said set of components in a hierarchy of dependencies during said descent.

23. (Original) The machine readable storage of claim 21, wherein said writing step further comprises the step of writing within said template at least one reference to a pre-requisite resource.

24. (Original) The machine readable storage of claim 21, wherein said listing step comprises the steps of:

accessing a template interface implemented by said set of components to permit access to content encapsulated by said set of components; and,

where a component within said set of components has not implemented said template interface, accessing said encapsulated content through a proxy to said component within said set of components.

25. (Original) The machine readable storage of claim 21, wherein said preserving step comprises the step of nesting component references in said listing in said template according to relative positions in said hierarchy of said set of components.

26. (Currently Amended) A machine readable storage having stored thereon a computer program for customizing an application, the computer program comprising a routine set of instructions for causing the machine to perform the steps of:

loading a template describing a customized configuration of the application

locating within said template a set of components arranged in a hierarchy of dependent components which are to be deployed in a customized configuration;

further locating within said template, a listing of points of variability for the variable establishment of configuration parameters for the customized application which can vary when ~~in each~~ deployment of the customized configuration;

deploying said set of components;

prompting an end user for values for said points of variability and applying said values to said points of variability; and,

returning control of the customized configuration to said end user.

27. (Original) The machine readable storage of claim 26, wherein said locating step comprises the steps of:

descending said hierarchy recursively; and,

deploying each of component in said hierarchy beginning with components in said set which are not dependent upon any other component in said hierarchy, and continuing through to at least one root component which is dependent upon all other components disposed below said at least one root component in said hierarchy.

28. (Original) The machine readable storage of claim 26, further comprising the steps of:

identifying within said template at least one reference to a pre-requisite resource; and,

deploying each pre-requisite resource identified within said template.

29. (Original) The machine readable storage of claim 26, wherein said deploying step comprises the steps of:

accessing a template interface implemented by said set of components to permit access to content encapsulated by said set of components; and,

where a component within said set of components has not implemented said template interface, accessing said encapsulated content through a proxy to said component within said set of components.